1. Django -- Football app -- TDD

Test -Driven Development (TDD) is a software development process relying on software requirements being converted to test cases before software is fully developed, and tracking all software development by repeatedly testing the software against all test cases. Below is the sequence of steps:

1. Add a "test"

2. Run all tests and see if the new test fails

3. Write the code

4. Run tests

5. Refactor code

In this challenge, you have to follow Test Driven Development Approach and add a new feature to the existing Football Tournament application.
The Product team has decided to add a new feature to the football tournament application. The unit test cases have already been designed for you and are available under the file named tests_new.py.

Your task for this challenge is to use TDD approach to build the new functionality in the application. Please ensure the existing functionality is not impacted by running regression tests.

The existing version of the application is named as alpha and the new version will be named beta. The views for the applications are named in the same way.

EXISTING FUNCTIONALITY:
In this application, There will be two users **admin** and **team**. A team can manage its own team after a successful login where as ADMIN can manage all the teams. The condition for the playing eleven is that in the **playing eleven** there must only **one goal keeper** , **atleast one mid-fielder**, **atleast one forwarder** and **atleast one defender.**
The **leaderboard** option which in this case is called mappings model is used to map teams for semi finals and final and declare winner. This leaderboard option is only updated by an admin user and there must be atleast 4 teams to manage leaderboard. Team can just view the leaderboard.

NEW FUNCTIONALITY:
In this feature update, you are going to work on a few more endpoints of the API to implement a new class of users named viewer who can **register, login, view matches mapped by admin, book tickets to a match and view all bookings made by the viewer.**

Try to finish the development of the new feature based on the unit tests written in file tests_new.py.

STEPS TO COMPLETE THE CHALLENGE:
1. Select **Install** from Project menu to install all the dependencies.
2. Select **Run** from Project menu to run the application.
3. Select **RUN TESTS** option to run the tests.

Software Instructions

This question requires **Python 2** and **Django 1.11.5**.

- [Install Python and Django](#)

Git Instructions

Use the following commands to work with this project

Run

```
python3 manage.py runserver 0.0.0.0:8000
```

Test

```
rm -rf unit.xml; python3 manage.py makemigrations && python3 manage.py migrate --run-syncdb && python3 manage.py test --keepdb
```

Install

```
sudo apt-get install libpq-dev python3-dev; pip3 install --user -r requirements.txt;
```